

COMPUTER SCIENCE

МОДИФІКАЦІЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ НА ОСНОВІ МЕТОДУ НЕЦЕНТРОВАНИХ ГОЛОВНИХ КОМПОНЕНТ ТА СТАНДАРТНІ ТЕСТИ**Шадура О. В.***Україна, м. Київ, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"***DOI:** https://doi.org/10.31435/rsglobal_ws/30042019/6464**ARTICLE INFO****Received:** 18 February 2019**Accepted:** 24 April 2019**Published:** 30 April 2019**KEYWORDS**

genetic algorithms, multiobjective optimization, principle component analysis, Pareto front, multiobjective functions.

ABSTRACT

The purpose of this article is to develop the necessary mathematical description of the method of the uncentered principal component analysis for the optimization of the genetic algorithm. A secondary goal is to evaluate the approximations for its application for HEP data analysis and to develop its program implementation for genetic algorithm together with a new operator based on the method of the uncentered principal components (UPCA-operator) and to check its efficiency on the example benchmark tests.

Citation: Shadura O. V. (2019) Modyfikatsiia Henetychnykh Alhorytmiv na Osnovi Metodu Netsentrovanykh Holovnykh Komponent ta Standartni Testy. *World Science*. 4(44), Vol.1. doi: 10.31435/rsglobal_ws/30042019/6464

Copyright: © 2019 Shadura O. V. This is an open-access article distributed under the terms of the **Creative Commons Attribution License (CC BY)**. The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Вступ. Для аналізу даних в експериментах на Великому адронному колайдері (LHC), розташованому в ЦЕРНі (Женева, Швейцарія) широко використовують ґрид-технології, які розподіляють обробку експериментальних даних серед великої кількості окремих обчислюваних кластерів, що побудовані з різних за структурою комп'ютерних платформ. Дослідження в області фізики високих енергій (ФВЕ) і ядерної фізики не можливі без використання великих обчислювальних потужностей та спеціального програмного забезпечення для обробки, моделювання та аналізу даних. Така ситуація обумовлена як великою кількістю даних, що генерується в експериментах на сучасних прискорювачах так і статистичною природою аналізу даних і складністю алгоритмів обробки даних які використовуються. Крім того, при комп'ютерній обробці даних необхідно моделювати умови роботи прискорювача, детекторів та фізичних процесів в детекторі одночасно с генерацією наборів даних та їх обробкою.

Нові можливості функціональності ядра GeantV [1] полягають в тому, щоб забезпечити формування структури даних та розпаралелювання сервісів для відправлення завдань та керування робочими потоками. Це можна реалізувати, забезпечивши когерентність роботи пакету шляхом пакування параметрів частинок у вектори та використання цих векторів для моделювання за SIMD-оптимізованими алгоритмами, враховуючи специфіку моделювання геометрії детектора та фізики транспорту частинок у детекторі.

Для налаштування роботи GeantV необхідне конфігурування інфраструктури керування подіями, що працює на нодах та забезпечуються а) матрицею налаштованих параметрів, яка включає: загальну кількість подій, кількість буферизованих подій; поріг для визначення пріоритетів подій; кількість потоків процесора; політику планування розміщення NUMA локальності; вихідний

розмір вектору; порогове значення між перевимкненням між скалярним та векторним режимами; спеціальний поріг під час активації кошика в диспетчері завдань; поріг пам'яті; поріг знищення подій та кількість пропагаторів, та б) характеристиками нодів, які можна оцінити за допомогою агрегованої матриці функцій пристосованості продуктивності, що включає: оптимізацію загального часу виконання симуляції та середній час оптимізації для створення однієї популяції; зменшення використання пам'яті додатків; оптимізацію кількості викликів інструкцій для моделювання та специфічне для поточної архітектури, зниження спорідненості пам'яті (NUMA оптимізація), споживаної мікропроцесором. Для дослідження оптимізації налаштування симуляцій у ФВЕ ми розглядаємо симуляцію GeantV, як задачу "чорної скриньки" з набором фітнес функцій. Згідно моделі GeantV, нам потрібно узагальнити набір цілей, які ми хотіли б досягти за допомогою процедури оптимізації та їхньої залежності від великої кількості векторних параметрів. Для цієї задачі вибираються головними наступні цільові функції:

- оптимізація загального часу виконання задачі симуляції та середній час оптимізації для створення однієї популяції;
- зменшення використання пам'яті додатків GeantV;
- оптимізація кількості викликів інструкцій для всього моделювання, специфічного для поточної архітектури;
- зниження спорідненості пам'яті, споживаної мікропроцесором, що використовується для моделювання симуляції GeantV;
- максимізувати операційну інтенсивність моделювання GeantV.

Стохастична оптимізація є одним з основних підходів в обчислювальній статистиці. Стохастична оптимізація означає мінімізацію (або максимізацію) цільових функцій за наявності випадковості в процесі оптимізації. Випадковість може бути пов'язана як із застосуванням методу Монте-Карло в процедурі моделювання, так із шумом у вимірюваннях або ж з обома випадками одночасно. Стохастичні методи оптимізації дуже зручні для оцінки цільових функцій за наявності шуму. В останнє десятиліття одними з найбільш перспективних та широкоживаних методів стохастичної оптимізації є генетичний алгоритми з недомінантним сортуванням - NSGA-II [2] та NSGA-III [3].

В цих алгоритмах після ініціалізації популяції, особи в популяції групуються в набір фронтів за принципом недомінантності. Перший фронт розглядають як абсолютно недомінантний набір у поточній популяції, на другому фронті переважають особи лише першого фронту і так далі. Для кожного окремого учасника на кожному фронті призначається оцінний ранг, що базується на номері фронту, до якого вони належать. Особинам у першому фронті надають рангове значення 1, особинам у другому -- значення 2 і так далі. Окрім значення рангу, вводять новий параметр, який називається щільністю скупчення, і його розраховують для кожної особи. Щільність скупчення означає, наскільки близько до своїх сусідів розташована особина. Більша середня щільність призводить до більшого розмаїття популяції. Батьків добирають з популяції за допомогою бінарних турнірів з урахуванням рангу та щільності скупчення. Між двома особинами добирається та, яка має менший ранг, або більшу щільність скупчення. За допомогою операторів кросовера та мутацій можна дібрати популяцію для генерування потомків. Популяцію повторно сортують за принципом недомінантності та до неї добирають лише найкращі N особин для створення нової популяції, де N - чисельність популяції.

В роботі [5] нами був запропонований додатковий генетичний оператор P (НГК-оператор), заснований на методі нецентрованих головних компонент (НГК), що був розроблений в [6,7]. Включення цього оператора в генетичний алгоритм призводить до більш швидкого наближення до околу істинного фронту Парето для задач багатокритеріальної оптимізації. Цей фронт визначає набір оптимальних розв'язків, що не домінують відносно один одного, та при переміщенні від однієї точки фронту Парето до іншої досягається певний програш в одній фітнес-функції та виграв в іншій. Парето фронт складається з ідеальних осіб популяції в генетичному алгоритмі, вибраних на основі відповідного набору параметрів оптимізації.

У випадку включення додаткового генетичного оператора P повний генетичний оператор $G_P(\vec{p})$ задається композицією чотирьох відображень:

$$G_P: \Lambda \rightarrow \Lambda, \quad G_P(\vec{p}) = P \circ C \circ U \circ F(\vec{p}). \quad (1)$$

Тут спеціальні генетичні оператори – селекції F , мутації U , кросоверу C , та НГК-оператор P діють в лінійному просторі: $\Lambda = (p_1, p_2, \dots, p_m)^t$ – векторів популяції, де компоненти p_α є ймовірністю знайти α -ту особину в генетичній популяції, яка складається з N різних особин. Генетичний оператор $G_\alpha(\vec{p})$ визначається ймовірністю появи особини α в наступній генерації нащадків якщо попередня популяція була \vec{p} .

Коротко нагадаємо основні положення методу нецентрованих головних компонент. Основним об'єктом в цьому методі є матриця даних розміру $m \times n$

$$\hat{X} = \{X_{\alpha,i}\} = \{(\vec{x}_\alpha)_i\} = \{\vec{x}_\alpha\}, \quad (2)$$

де $\vec{x}_\alpha = \{(\vec{x}_\alpha)_i\} (1 \leq \alpha \leq m, 1 \leq i \leq n)$ є α -тою особиною в популяції. В цій матриці індекс i нумерує параметри, які вибрані для оптимізації використання програмного додатку GeantV ($i = 1, \dots, n$) та індекс α нумерує кількість експериментальних вимірювань для цільової функції ($\alpha = 1, \dots, m$ для m вимірювань). В термінах генетичних алгоритмів матриця даних описується через m вибірок даних в n -вимірному просторі де n є розмірність вектора, що описує особини та m є кількість особин в поколінні.

Головним об'єктом в методі НГК є матриця нецентральних других моментів

$$\hat{T} = \frac{1}{m} \hat{X}^t \cdot \hat{X}. \quad (3)$$

Ця матриця має ортонормовані власні вектори \vec{w}_j з відповідними власними значеннями t_j

$$\hat{T} \cdot \vec{w}_j = t_j \cdot \vec{w}_j, \quad \vec{w}_i^t \cdot \vec{w}_j = \delta_{i,j}, \quad 1 \leq i, j \leq n.$$

Власний вектор \vec{w}_j визначає j -ту «нецентровану» головну компоненту $\vec{\theta}_j = \{(\theta_j)_\alpha\} = \{\vec{x}_\alpha \cdot \vec{w}_j\} (\alpha = 1, \dots, m)$. Визначимо матрицю нецентрованих головних компонент $\Theta_{\alpha,j} = \{\vec{\theta}_j\}_\alpha = \{(\theta_j)_\alpha\}$, яку можна виразити через матрицю даних:

$$\Theta_{\alpha,j} = X_{\alpha,i} W_{i,j}, \quad \Theta_{i,\alpha}^t \Theta_{\alpha,j} = m \Delta_{i,j} = m t_j \delta_{i,j}, \quad 1 \leq \alpha \leq m. \quad (4)$$

Зручно ввести ортогональну матрицю нецентрованих головних компонент $\tilde{\Theta}_{\alpha,j}$ наступним чином:

$$\Theta_{\alpha,j} = \sqrt{m} \tilde{\Theta}_{\alpha,i} \Delta_{i,j}^{1/2}, \quad \Delta_{i,j}^{1/2} = t_i^{1/2} \delta_{i,j}, \quad \tilde{\Theta}_{i,\alpha}^t \tilde{\Theta}_{\alpha,j} = \delta_{i,j}. \quad (5)$$

За допомогою цієї матриці нескладно отримати сингулярне представлення для нецентрованої матриці даних

$$X_{\alpha,i} = \sqrt{m} \tilde{\Theta}_{\alpha,k} \Delta_{k,j}^{1/2} W_{j,i}^t. \quad (6)$$

Коли матриця нецентрованих других моментів \hat{T} має $(n-q)$ найменших власних значень $t_j \ll 1, q+1 \leq j \leq n$, то можна використати наближення, що контролюється малістю параметра власних значень, і отримати наближену матрицю даних $\tilde{X}_{\alpha,i}$ рангу q

$$\tilde{X}_{\alpha,i} = \sqrt{m} \tilde{\Theta}_{\alpha,k} \tilde{\Delta}_{k,j}^{1/2} W_{j,i}^t = \sqrt{m} (t_1^{1/2} \tilde{\Theta}_{\alpha,1} W_{1,i}^t + \dots + t_q^{1/2} \tilde{\Theta}_{\alpha,q} W_{q,i}^t), \quad (7)$$

де матриця власних значень $\tilde{\Delta}_{k,j}$ має ранг q ($t_{q+1} = t_{q+2} = \dots = t_n = 0$). Тоді можна наблизити матрицю $X_{\alpha,i}$ з рангом n за допомогою матриці $\tilde{X}_{\alpha,i}$ з рангом q . Ця процедура визначає додатковий генетичний оператор P (НГК-оператор), заснований на методі нецентрованих головних компонент (НГК).

Нескладно оцінити середньоквадратичну похибку η_q для наближення

$$\eta_q = \frac{1}{mn} \sum_{\alpha=1}^m \sum_{i=1}^n (X_{\alpha,i} - \tilde{X}_{\alpha,i})^2 = \frac{1}{n} \sum_{k=q+1}^n t_k.$$

Похибка є мінімальною, коли матриця нецентральных других моментів \hat{T} має $(n-q)$ найменших власних значень $t_j \ll 1$, $q+1 \leq j \leq n$.

Тести. Тести є одним з поширених способів перевірки коректності роботи генетичних алгоритмів, вони дають змогу порівняти ефективність різних генетичних алгоритмів, їх вихідний розподіл, статистику і провести аналіз роботи операторів в генетичних алгоритмах та загальний аналіз продуктивності стохастичного алгоритму. Простими прикладами таких тестів є стандартний набір DTLZ-тестів (Deb, Thiele, Laumanns, Zitzler) для багатоцільових задач оптимізації [4].

Тести DTLZ сконструйовані таким чином, що багатоцільові задачі оптимізації можуть масштабуватися до будь-якої кількості функцій. Крім того, всі DTLZ1-DTLZ6 масштабуються по відношенню до кількості параметрів, але мають фіксовану кількість вихідних параметрів $M - 1$, де M – кількість цільових функцій. Ці вихідні параметри є взаємопов'язаними, оскільки спроба оптимізувати їх за одним параметром одночасно (лише за один прохід) не буде визначати всі глобальні оптимуми. Відзначимо деякі особливості цих тестів: що цільові функції в DTLZ1-DTLZ4 мають кілька глобальних оптимумів. DTLZ5 та DTLZ6 є тестовими задачами з виродженими оптимальними фронтами Парето, де оптимальні fronti Парето – це дуги, вкладені в M -розмірний об'єкт.

При застосуваннях багато-цільових генетичних алгоритмів до задач багато-критеріальної оптимізації їх ефективність перевіряється рішенням двох задач: а) спроможністю еволюційного алгоритму давати збіжність до Парето-оптимального фронту ("задача збіжності") та б) давати гарний розподіл оптимальних розв'язків по всьому Парето фронту ("задача поширення").

Для перевірки ефективності алгоритму для рішення задачі збіжності, тестова задача повинна мати перепони для еволюції системи, наприклад, у вигляді великої кількості локальних парето-оптимальних фронтів. Для перевірки ефективності рішення задачі поширення, тестова задача повинна мати області ущільнення оптимальних розв'язків на Парето фронті, наприклад, за рахунок існування локальних парето-оптимальних атракторів на цьому фронті. Цього можна досягнути роблячи парето-оптимальні fronti неопуклими, дискретними і такими, що мають змінну щільність оптимальних рішень вздовж фронту.

Набір тестів DTLZ дозволяє досліджувати ефективність методів рішення багатоцільових задач оптимізації контрольованим чином, з відомими характеристиками, кількістю параметрів та знанням оптимального фронту Парето для кожного тесту.

Для знаходження оптимальних значень параметрів при оптимізації симуляцій GeantV ми розглядаємо модель GeantV як задачу "чорної скриньки" з скінченим набором фітнес функцій. З огляду на це, чим більше ми знайдемо оптимальних значень параметрів в околі Парето-оптимального фронту тим простіше буде реалізувати такі оптимальні рішення для обчислювальної інфраструктури що використовується для поточної обробки даних за допомогою GeantV. Тоді для перевірки ефективності застосованого генетичного алгоритму NSGA-II з включенням НГК-оператора для оптимізації роботи GeantV необхідно використовувати такі тестові задачі які дозволяють перевірити алгоритм на отримання широкого розподілу оптимальних розв'язків в околі всього Парето фронту ("задача поширення"). Серед тестів DTLZ для перевірки ефективності рішення задачі поширення зручними є тести DTLZ2 та DTLZ4.

З цієї точки зору в статті ми обмежуємось перевіркою ефективності запропонованої нами модифікації генетичних алгоритмів за умови включення в них НГК-оператора на основі тестових задач DTLZ2 та DTLZ4 та порівняння ефективності цієї модифікації з роботою генетичних алгоритмів без НГК-оператора. Давайте коротко опишемо кожну з цих тестових задач.

Тестова задача DTLZ2. Ця тестова задача є M -цільовою задачею оптимізації з випуклим оптимальним фронтом Парето (маємо знайти спільні оптимуми M функцій, $m=1, \dots, M$):

$$\text{minimize: } f_m(\vec{x}) = (1 + g(\vec{z})) \cdot \prod_{l=1}^{M-m} \cos\left(\frac{\pi}{2} x_l\right) \cdot \sin\left(\frac{\pi}{2} x_{M-m+1}\right) \quad (2)$$

де

$$g(\vec{z}) = \sum_{i=1}^n (z_i - 0.5)^2.$$

Тут $\vec{x} = (x_1, x_2, \dots, x_{M-1})$, цільові функції $f_l(\vec{x})$ ($l = 1, \dots, M$) залежать від n параметрів $\vec{z} = (z_1, z_2, \dots, z_n)$, причому змінні x_l лежать в інтервалі $0 \leq x_l \leq 1$ ($l = 1, \dots, M - 1$) і

параметри z_i – в інтервалі $0 \leq z_i \leq 1$ ($i = 1, \dots, n$). Зауважимо, що ці інтервали для змінних \vec{x} та параметрів \vec{z} будуть використані також і для тестової задачі DTLZ4.

Парето-оптимальний розв'язок для DTLZ2 відповідає значенням параметрів $\vec{z}^* = (0.5, \dots, 0.5)$ та оптимальні значення цільових функцій розташовані на гіперповерхні:

$$\sum_{l=1}^M (f_l^*)^2 = 1.$$

Для тестування ми вибрали число параметрів $n = 10$, яке часто використовується в літературі для цього випадку. Парето-оптимальний фронт знайдений за допомогою генетичного алгоритму NSGA-II зображений на Рис.1. (для зручності використали $M = 3$).

Тестова задача DTLZ4. Ця тестова задача є M -цільовою задачею оптимізації для перевірки ефективності генетичного алгоритму для рішення "задачі поширення" оскільки має локальні ущільнення оптимальних розв'язків на Парето фронті (маємо знайти спільні оптимуми M функцій, $m=1, \dots, M$):

$$\text{minimize: } f_m(\vec{x}) = (1 + g(\vec{z})) \cdot \prod_{l=1}^{M-m} \cos\left(\frac{\pi}{2} x_l^\alpha\right) \cdot \sin\left(\frac{\pi}{2} x_{M-m+1}^\alpha\right) \quad (4)$$

де

$$g(\vec{z}) = \sum_{i=1}^n (z_i - 0.5)^2.$$

Зафіксуємо значення параметра $\alpha=100$ та параметра $n=10$, як було вибрано в статті [4]. Ця багато-цільова задача має щільну множину рішень в околі $f_M - f_1$ площини. Використовуючи генетичний алгоритм NSGA-II цей результат отримано після 200 генерацій поколінь при $M = 3$ і показано на малюнку 2. В цьому підході кінцеві покоління дуже корелюють з вибором початкових поколінь і в роботі [4] було отримано три варіанти ущільнення рішень: а) оптимальні рішення знаходяться в площині $f_3 - f_1$, б) оптимальні рішення знаходяться в площині $f_1 - f_2$, в) рішення заповнюють всю парето-оптимальну поверхню.

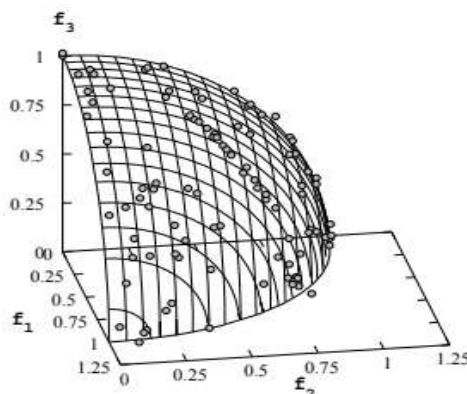


Рис. 1. Парето фронт для тесту DTLZ2 (NSGA-II).

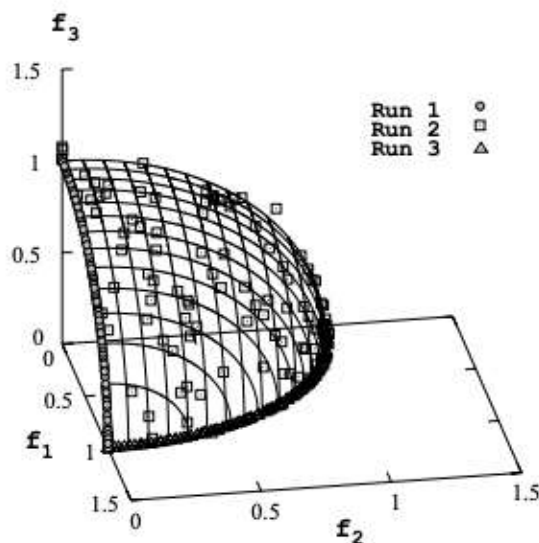


Рис. 2. Парето фронт для тесту DTLZ4 (NSGA-II).

Використовуючи ці тести був проведений порівняльний аналіз використання різних варіантів включення НГК-оператора в генетичний алгоритм для центрованих і нецентрованих даних за допомогою оцінки середнє квадратичної похибки і показано, що наступна комбінація NSGAII та методу нецентрованих головних компонент є найбільш ефективним для збіжності алгоритму до Парето фронту:

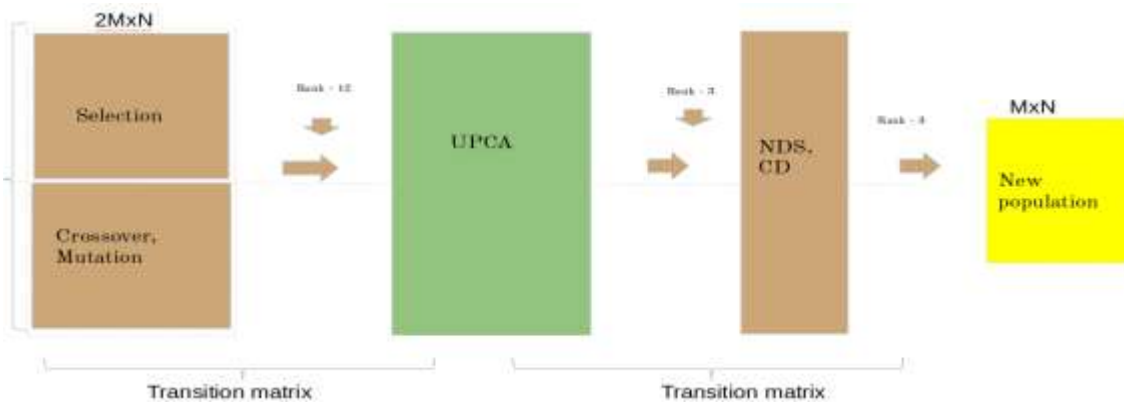


Рис. 3. Інтеграція нового оператора в генетичний алгоритм NSGA-II.

Наступним важливим кроком є перевірка того, що алгоритм є обчислювально ефективним. У NSGA-II (Рисунок 3.) батьків добирають з популяції за допомогою бінарних турнірів з урахуванням рангу та параметру щільності скупчення індивідумів. Між двома особинами добирається та яка має менший ранг або більшу щільність скупчення. За допомогою операторів кросовера та мутацій можна сформувати популяцію для генерування нащадків. Наступним кроком використовується НГК-оператор. Популяцію повторно сортують за принципом не домінантності та до неї добирають лише найкращі N особин для створення нової популяції, де N - чисельність популяції.

Результати. Тести DTLZ [4] являють собою набір чисельних мультиоб'єктивних задач, які використовуються для порівняння / валідації алгоритмів GA. Ми представляємо на виході тестування тестів DTLZ як порівняння NSGA-II без і з оператором очищення шуму UPCA.

В статті зроблене порівняння результатів застосування алгоритмів NSGA-II [2] та NSGA-II з включенням НГК-оператора для тестів DTLZ. Відмітимо, що ефективність алгоритму NSGA-III перевищує NSGA-II, але для простоти ми перевірили наш метод використовуючи алгоритм NSGA-II і отримані результати показують ефективність нашого підходу. На Рис.4, 5 представлені розподіли параметрів (середнє та стандартне значення відхилення (дисперсію)) та поведінку значення вартості залежно від використовуваних алгоритмів.

Порівнюючи Рис.5, на якому приведений розподіл населення 10-того покоління для тестів DTLZ2 та DTLZ4 з застосування НГК-оператора, з Рис.4, де приведений розподіл населення 10-того покоління для тестів DTLZ2 та DTLZ4 без застосування НГК-оператора, можна спостерігати швидку збіжність до ідеальних значень параметрів задачі у першому випадку (при наявності НГК-оператора). Рис.5 показує наближення до фронту Парето у поєднанні з правильним набором параметрів.

Для нового алгоритму протестованого разом з новим оператором, ми пропонуємо перші попередні результати DTLZ тестів:

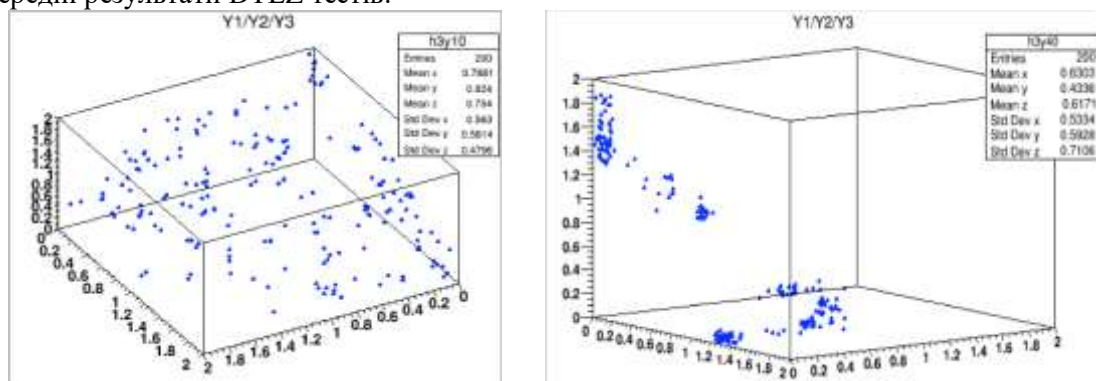


Рис. 4. Генетичний алгоритм NSGA-II для тестів DTLZ2 та DTLZ4.

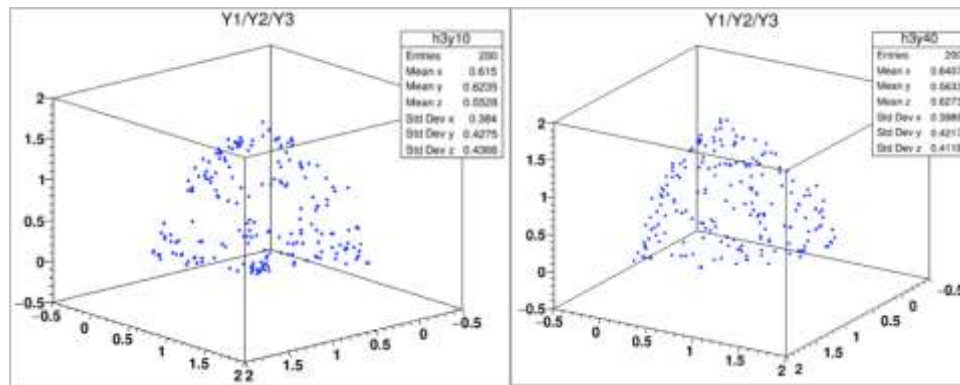


Рис. 5. Результати після включення НГК-оператора в генетичний алгоритм NSGA-II для тесту DTLZ2 та DTLZ4.

Висновки. У цій статті були приведені результати тестування нового НГК-оператора при застосуванні його в стандартних тестових багато-цільових задачах для генетичних алгоритмів. Проведено шляхом моделювання порівняльний аналіз використання різних варіантів введення методу головних компонент в генетичний алгоритм NSGA-II та за допомогою оцінки середньоквадратичної похибки знайдено оптимальний спосіб включення НГК-оператора (показаний на Рисунок 3) який виявився найефективнішим для конвергенції алгоритму до фронту Парето.

ЛІТЕРАТУРА

1. G. Amadio and A. Ananya and J. Apostolakis and A. Arora and M. Bandieramonte and A. Bhattacharyya and C. Bianchini and R. Brun and P. Canal and F. Carminati and L. Duhem and D. Elvira and A. Gheata and M. Gheata and I. Goulas and R. Iope and S. Jun and G. Lima and A. Mohanty and T. Nikitina and M. Novak and W. Pokorski and A. Ribon and R. Sehgal and O. Shadura and S. Vallecorsa and S. Wenzel and Y. Zhang, GeantV: from CPU to accelerators, Journal of Physics: Conference Series, 762, 1, p.012019, 2016
2. K. Deb and A. Pratap and S. Agarwal and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, 2002, Vol.6, p.182-197, Doi:10.1109/4235.996017
3. Seada, Haitham, and Kalyanmoy Deb. "U-NSGA-III: A Unified Evolutionary Optimization Procedure for Single, Multiple, and Many Objectives", IEEE Trans. Evolutionary Computation 20(3): 358-369 (2016)
4. K. Deb and L. Thiele and M. Laumanns and E. Zitzler, Scalable Test Problems for Evolutionary Multi-Objective Optimization, Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, Springer, 2005
5. Shadura O. Multivariate convergence-targeted operator for the genetic algorithm / O. Shadura, A. Petrenko, S. Svistunov // Системні дослідження та інформаційні технології: міжнародний науково-технічний журнал, № 1. с. 126–140 (2017).
6. Шадура О.В. Метод головних компонент і оптимізація пакетів фізичного моделювання за допомогою генетичних алгоритмів, Вісник Університету «Україна», Серія «Інформатика, обчислювальна техніка та кібернетика», №1(22), с.198-209 (2019).
7. Oksana Shadura, Federico Carminati and Anatoliy Petrenko. Performance Optimization of Physics Simulations Through Genetic Algorithms, Journal of Computer Science, v.15, Issue 1, p. 57-66 (2019) (DOI 10.3844/jcssp.2019.57.66)